Simulation Model and Technology Evaluation Test-bed (SMTET)

User's Manual

The scope of SMTET is the development of a state-of-the-art, high-fidelity simulation model for the Port of Savannah with a 3-dimensional, true-to-scale animation.

Off-the-shelf simulation products incorporate impressive animation capabilities, but often place severe constraints on implementing various modeling aspects. These limitations, along with the desire for flexibility and ownership of the test-bed, motivated the development of an engine based on a publicly available, web-enabled programming language with object-oriented capabilities, such as Java. Amongst the various Java-based simulation packages, we chose the Distributed Simulation Object Library (DSOL, ver. 2.0) as the backbone because it has been used to develop large-scale, distributed simulation models for port operations (e.g., the Port of Rotterdam in The Netherlands). DSOL is a full-featured discrete-event simulation environment developed at Delft University of Technology, The Netherlands, and is available at http://sourceforge.net/projects/dsol/.

The animation is an extension to the existing DSOL-based system. It is likewise implemented using Java and utilizes the cross-platform Java 3D and Java Advanced Imaging API (Application Programmer Interface) for graphics. It presents a 3D view of the same information presented by the existing 2D visualization. However, it is improved with better usability and is easier to interpret because of the use of realistic 3D imagery. On the backend, the 3D animation system interfaces with the DSOL system using the API to exchange detailed information about the state of the simulation. It keeps track of all state changes, allowing for playback at different speeds and times (see Section 2). Scaled models of all mobile entities are loaded from the commonly used LightWave OBJ format and allow for detailed customization as necessary (DXF or other formats can be converted).

The test-bed enables various groups such as emergency management agencies, military logistics planners, and Port management to: (a) create and test alternative operational strategies, sensor technologies (e.g., RFID), and various other applicable technologies; and (b) utilize the SMTET to conduct disaster preparedness or other emergency situation training that is neither possible nor practical to do at the facility itself. Further, the test-bed allows users to train or upgrade employees on new software for Port operations. One example of this is the need to train new employees on the terminal operating software in use at the Port. The employees could attend an in-depth training class at the Logistics Innovation Center (LINC) and get real-world situational training for the same system they will be using at work.

The current model was built based using a map and dimensional measurements taken from Google Earth. The files are currently located at the site http://www.isye.gatech.edu/~ywu60/smtet/, but will be moved to a powerful Linux server soon. The site contains the Java Web Start file smtet.jnlp, and the scenario file, currently SavannahXmlMap.xml. The application can also be launched directly from the site. Under Mac OS X, Java Web Start installs an application (*.app file) in the Applications directory.

1 Requirements

- Operating System: Windows (XP, Vista or 7), Macintosh (OS X 10.5 or higher), Linux, and Unix.
- Software Requirements:
  - Version of Java JRE 1.5.
  - OpenGL, ver. 1.2 or later.
- Hardware Requirements:
  - 3.0 GB of RAM.
  - Dedicated graphics card with at least 256 MB of memory.

2 Running the Application

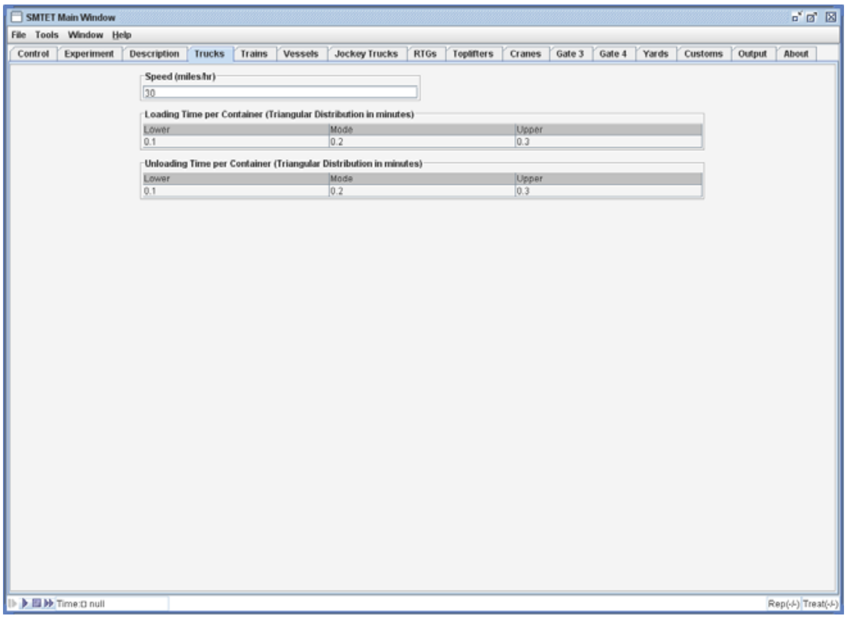
The input parameters for the simulation run can be assigned by creating an xml file with structure similar to *SavannahXmlMap.xml* of by using the tabs in the application's Main Window, as illustrated in the following headings. The scenario file is selected from the **File > Open File** menu in the Main Window. Or you could select the embedded scenario file from the **Help > Tutorial > SavannaXmlMap.xml** menu in the Main Window.

2.1 Control Tab

This tab allows the user to input the run length, the number of independent replications, the time step, and the animation delay.

2.2 Trucks Tab

This tab allows the user to input the speed and number of trucks, and parameters for the triangular distributions for container loading and unloading times (see Figure 1). The tabs for the Jockey Trucks, the Rubber Tired Gantry (RTG) cranes, the Toplifters, and Cranes are similar.


SMTET Main Window
File Tools Window Help
Control Experiment Description Trucks Trains Vessels Jockey Trucks RTGs Toplifters Cranes Gate 3 Gate 4 Yards Customs Output About
Time:□ null
Rep(-/-) Treat(-/-)

Speed (miles/hr)

30

Loading Time per Container (Triangular Distribution in minutes)

Lower	Mode	Upper
0.1	0.2	0.3

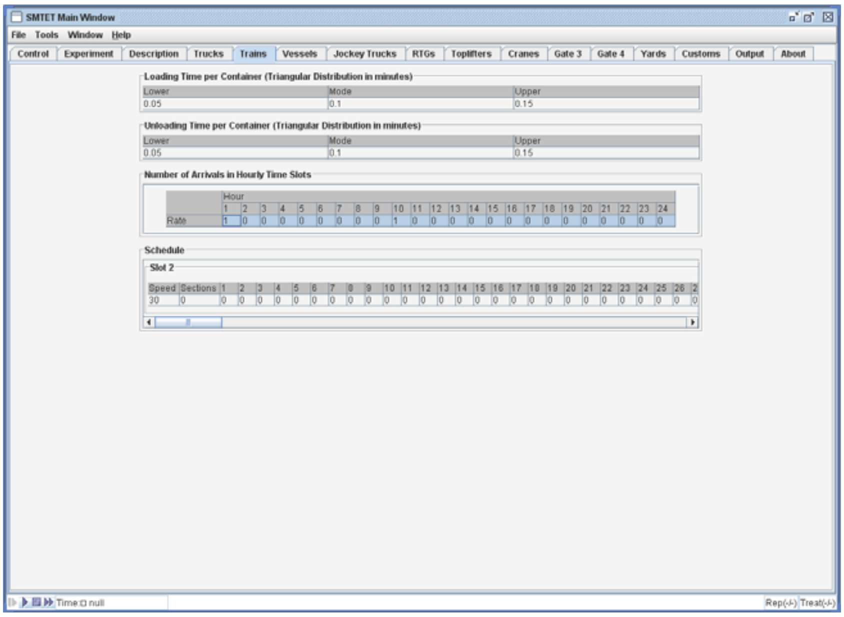
Unloading Time per Container (Triangular Distribution in minutes)

Lower	Mode	Upper
0.1	0.2	0.3

Figure 1: Data for Trucks

2.3 Trains Tab

This tab is used for entering information related to the schedule of arriving trains, their contents, and the unloading/loading times per container (Figure 2). The latter times are modeled using the triangular distribution. The day is divided into hourly slots. The number of arrivals in each slot should be 0 or 1. Clicking in a time slot activates a dialog where the user can enter the speed of the train, the number of rail sections, and the number of containers in each section.


SMTET Main Window
File Tools Window Help
Control Experiment Description Trucks Trains Vessels Jockey Trucks RTGs Toplifters Cranes Gate 3 Gate 4 Yards Customs Output About
Time: null
Rep(-/-) Treat(-/-)

Loading Time per Container (Triangular Distribution in minutes)

Lower	Mode	Upper
0.05	0.1	0.15

Unloading Time per Container (Triangular Distribution in minutes)

Lower	Mode	Upper
0.05	0.1	0.15

Number of Arrivals in Hourly Time Slots

	Hour																							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Rate	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Schedule

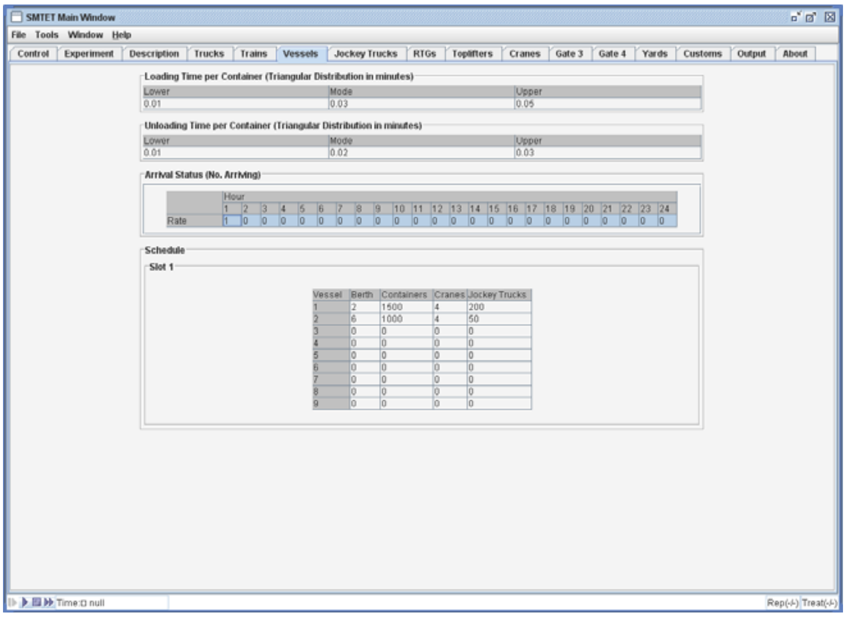
Slot 2

Speed	Sections	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	2
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 2: Rail Data

2.4 Vessels Tab

This tab is used for entering information related to the schedule of arriving vessels, their contents, and the unloading/loading times per container (Figure 3). The latter times are modeled using the triangular distribution. The day is divided into hourly slots. Clicking in a time slot activates a dialog where the user can enter the berth for each vessel, the number of containers, and the numbers of cranes and jockey trucks assigned to the vessel.



Loading Time per Container (Triangular Distribution in minutes)

Lower	Mode	Upper
0.01	0.03	0.05

Unloading Time per Container (Triangular Distribution in minutes)

Lower	Mode	Upper
0.01	0.02	0.03

Arrival Status (No. Arriving)

	Hour																							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Rate	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Schedule

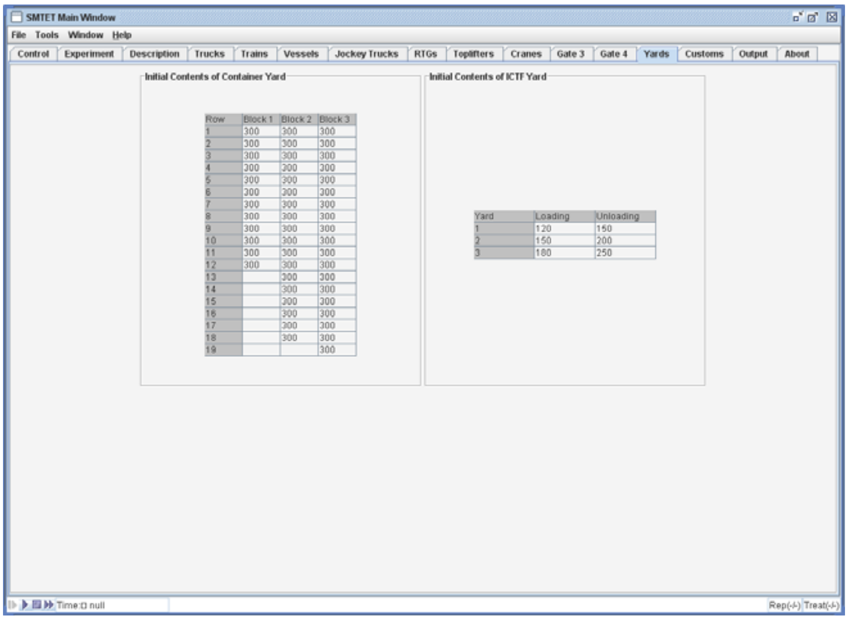
Slot 1

Vessel	Berth	Containers	Cranes	Jockey Trucks
1	2	1500	4	200
2	6	1000	4	50
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0

Figure 3: Vessel Data

2.5 Yards Tab

This tab allows the user to input the initial contents of each container yard (by row and block) as well as the yard at the Intermodal Container Transfer Facility (ICTF). See Figure 4.



Initial Contents of Container Yard

Row	Block 1	Block 2	Block 3
1	300	300	300
2	300	300	300
3	300	300	300
4	300	300	300
5	300	300	300
6	300	300	300
7	300	300	300
8	300	300	300
9	300	300	300
10	300	300	300
11	300	300	300
12	300	300	300
13		300	300
14		300	300
15		300	300
16		300	300
17		300	300
18		300	300
19			300

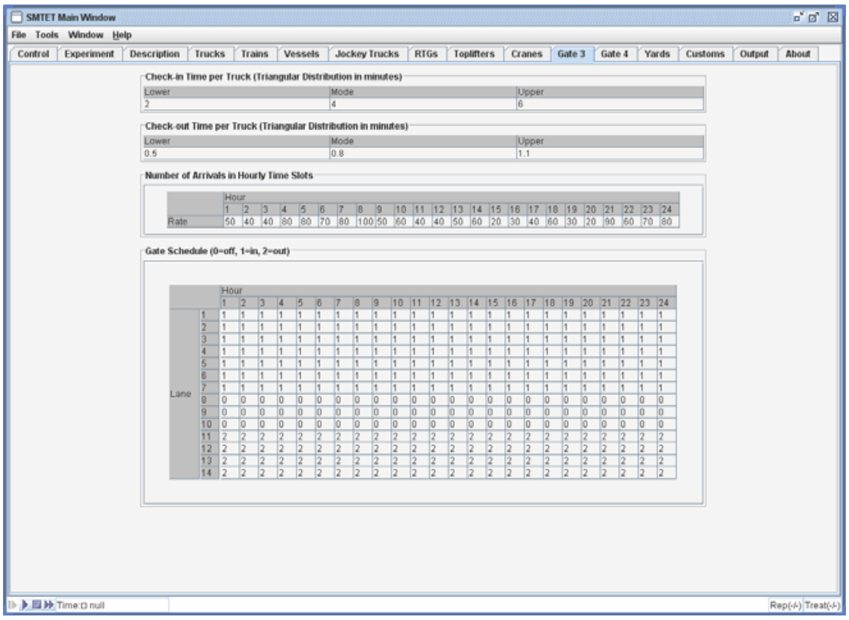
Initial Contents of ICTF Yard

Yard	Loading	Unloading
1	120	150
2	150	200
3	180	250

Figure 4: Initial Yard Contents

2.6 Gate Tabs

The gate tabs allow the user to input the operational parameters for Gates 3 and 4. The check-in and check-out times per truck follow triangular distributions. The arrival processes for trucks at each gate follow the nonstationary Poisson model with rates that can vary on an hourly basis. (Under this model, the number of arrivals during a given time window follow the Poisson distribution with mean equal to the area under the rate function during that time window. Finally, the user can designate each lane as ingoing (1), outgoing (2) or nonfunctional (0).


SMTET Main Window
File Tools Window Help
Control Experiment Description Trucks Trains Vessels Jockey Trucks RTGs Toplifters Cranes Gate 3 Gate 4 Yards Customs Output About
Time: null
Rep(-/-) Treat(-/-)

Check-in Time per Truck (Triangular Distribution in minutes)

Lower	Mode	Upper
2	4	6

Check-out Time per Truck (Triangular Distribution in minutes)

Lower	Mode	Upper
0.5	0.8	1.1

Number of Arrivals in Hourly Time Slots

	Hour																							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Rate	50	40	40	80	80	70	80	100	50	60	40	40	50	60	20	30	40	60	30	20	90	60	70	80

Gate Schedule (0=off, 1=in, 2=out)

		Hour																							
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Lane	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	11	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	12	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	13	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	14	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

Figure 5: Tab for Gate 3

2.7 Customs Tab

This tab is used for setting the fraction of containers undergoing customs inspection and the parameters of the triangular distribution for the inspection time per container. This inspection is not displayed in the current animation; it is modeled only as delay.

2.8 Model Execution

To run the model, hit the Play button at the bottom (left) corner of the application window. The 3D animation window is activated by the **Window > 3D Animation** menu item (or the **F5** key).

The animation is onset by clicking on the Play button on the 3D Animation window. The following controls are available to the user:

- Time Controls:
    - Play button: Start the animation.
    - Time Step: While the Play button is in Pause mode, type the desired time (past or future not to exceed the simulation time) in the time indicator. Alternatively, use the time slide to move to the desired time.
    - Time Scale: This slide is used to control the speed of the animation (independently of the speed of simulation).
- Location Controls: The animation perspective can be changed by means of the following controls:
    - Moving the mouse pointer.
    - Zoom in or out: Use mouse wheel.

- Move forward: W
- Move backward: S
- Move left: A
- Move right: D
- Orbit left (clockwise): Q
- Orbit right (anti clockwise): E
- Orbit down (grazing angle): Z
- Orbit up: X

Note: Holding the shift key will increase the speed of movement/rotation.

Figure 6 below depicts a vessel along with the assigned cranes, a nearly container yard, and several Rubber Tired Gantry (RTG) cranes.

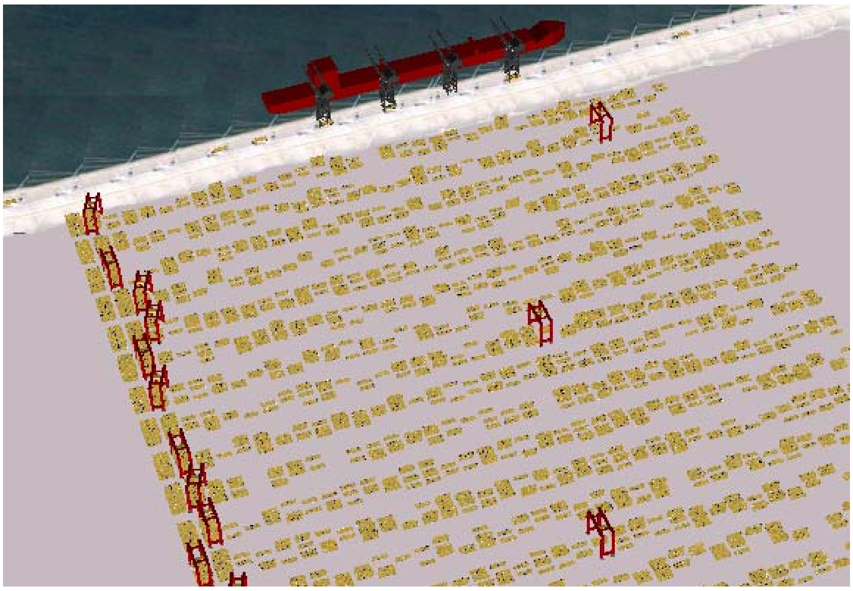
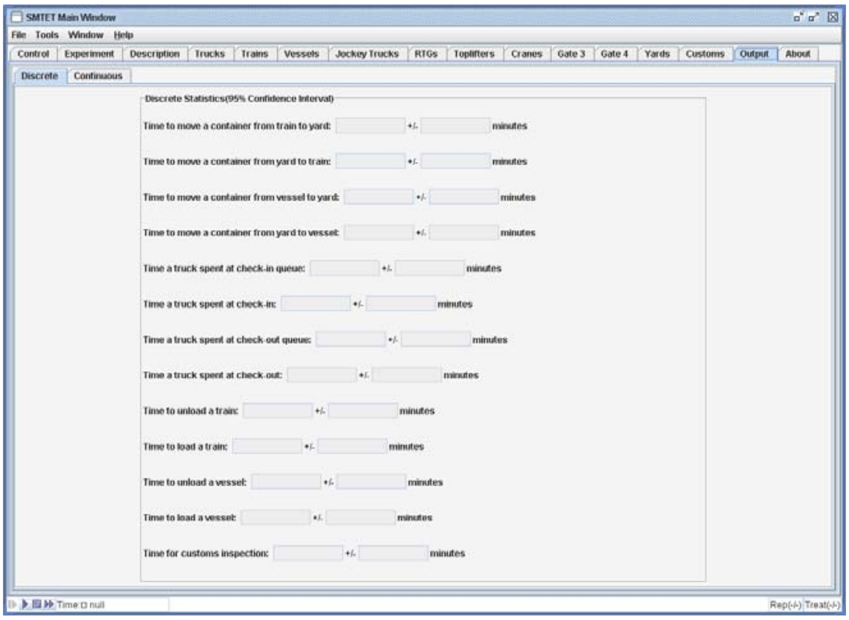


Figure 6: A Vessel and a Nearby Container Yard

3 Output Statistics

Tab Output contains point estimates and 95% confidence intervals for various means estimated by means of discrete- and continuous-time statistics (Figures 7-8). Confidence intervals are generated only when the number of replications exceeds one, but they practically require several replications to be meaningful.


SMTET Main Window
File Tools Window Help
Control Experiment Description Trucks Trains Vessels Jockey Trucks RTGs Toplifters Cranes Gate 3 Gate 4 Yards Customs Output About
Discrete Continuous
Time: null
Rep(-/-) Treat(-/-)

Discrete Statistics(95% Confidence Interval)

Time to move a container from train to yard: +/- minutes

Time to move a container from yard to train: +/- minutes

Time to move a container from vessel to yard: +/- minutes

Time to move a container from yard to vessel: +/- minutes

Time a truck spent at check-in queue: +/- minutes

Time a truck spent at check-in: +/- minutes

Time a truck spent at check-out queue: +/- minutes

Time a truck spent at check-out: +/- minutes

Time to unload a train: +/- minutes

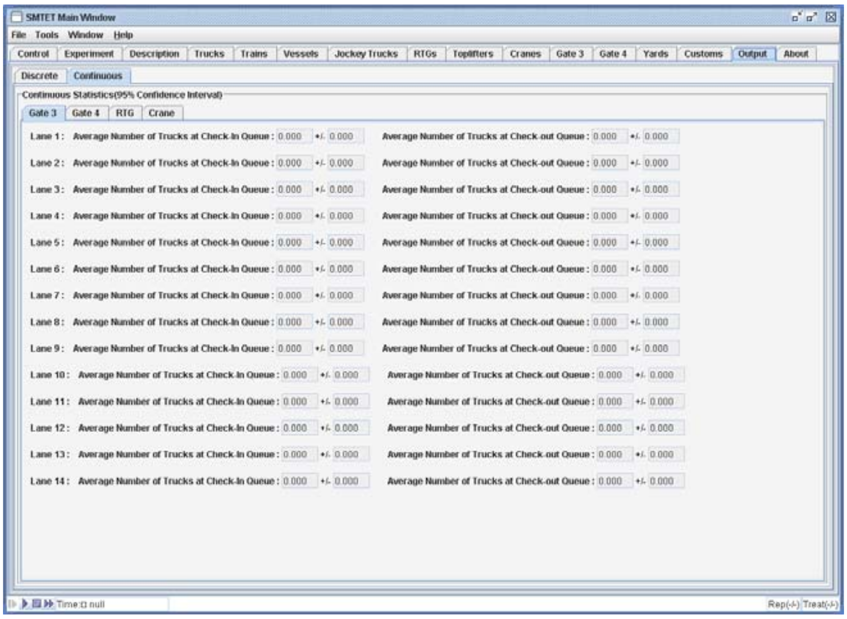
Time to load a train: +/- minutes

Time to unload a vessel: +/- minutes

Time to load a vessel: +/- minutes

Time for customs inspection: +/- minutes

Figure 7: Discrete-time Averages and Confidence Intervals


SMTET Main Window
File Tools Window Help
Control Experiment Description Trucks Trains Vessels Jockey Trucks RTGs Toplifters Cranes Gate 3 Gate 4 Yards Customs Output About
Discrete Continuous
Gate 3 Gate 4 RTG Crane
Time: null
Rep(-/-) Treat(-/-)

Continuous Statistics(95% Confidence Interval)

Lane 1 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 2 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 3 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 4 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 5 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 6 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 7 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 8 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 9 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 10 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 11 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 12 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 13 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Lane 14 : Average Number of Trucks at Check-In Queue : 0.000 +/- 0.000 Average Number of Trucks at Check-out Queue : 0.000 +/- 0.000

Figure 8: Continuous-time Averages and Confidence Intervals

Additional statistics and graphs can be viewed from the **Window > Statistics** menu item (**F2** key). This action opens a new window with a customizable number of rows and columns. Each entry of the window can be filled with graphs of estimates by dragging contents from the left (single) column to the window.

4 Appendix

4.1 Parameters

a) **experiment:**

- **name**: Name of the experiment.
- **Model:** Location of the Main class of the model. This parameter should not be changed since it is the location of the Main class in the jar file. The value is *gatech.isye.sim.smtet.scenario.Scenario.*
- **classPath**:
  - jar-file: Location of the jar file. This parameter shall have the location of the model jar file, (e.g. <jar-file>file:/D:/Georgia Tech/Javajars/Savannah-0.99.jar</jar-file>).

    Note: Java uses forward slash (/) instead of backslash (\) characters to specify the path.

b) **treatment**

- **startTime:** Start time for the simulation. (Not used in the current version.)
- **TimeUnit:** Time units for the simulation. This should be kept in seconds, as for the time being some model parameters are included in the code in this time unit.
- **warmupPeriod:** System statistics will be discarded at the end of this time period.
- **runLength:** Duration of a run.
- **numberofreplication:** This parameter is a DSOL parameter not used in our project.
- **properties:** Properties are variables that are fixed for each treatment (experiment). These variables have the following sequence of elements. Only one of these fields is currently read by the Port of Savannah model.

  Key : simulation.totalreplications

  Value: The **number** of replications to be run.

  Key : simulation.tracetofile

  Value: True or false. This parameter defines whether a log file is created or not.

  Key : simulation.enhanced-animation

  Value: True or false. This parameter defines whether enhanced animation can be played from the animation menus.

  Key : simulation.enhanced-animation.class

  Value: .gatech.ece.savannah.Animation3D. The user should not change this value since it specifies the name and location of the Java class file that will run the enhanced animation.

  Key : simulation.map-file

  Value: Name of the XML file that defines the port lay-out.

4.2 XSD file

The following is the schema verification (**xsd**) file that is used to check the parameter file for consistency.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema
          xmlns:xsd="http://www.w3.org/2001/XMLSchema"
          xmlns:smtet="http://www.isye.gatech.edu"
          targetNamespace="http://www.isye.gatech.edu">
<!-- ################################ DEFINITION OF DATABASE ######################### -->
          <xsd:element name="simulationparametersdatabase">
                    <xsd:complexType mixed="true">
                              <xsd:sequence>
                                        <xsd:element maxOccurs="unbounded" minOccurs="1"
                                                  name="experimentDef" type="smtet:experimentalFrame"/>
                              </xsd:sequence>
                    </xsd:complexType>
          </xsd:element>
<!-- ################################### TYPE DEFINITIONS ####################################################### -->
          <xsd:complexType name="experimentalFrame" mixed="true">
                    <xsd:sequence>
                              <xsd:element maxOccurs="unbounded" minOccurs="1" name="experiment" type="smtet:experiment"/>
                              </xsd:sequence>
          </xsd:complexType>
          <xsd:complexType name="experiment" mixed="true">
                    <xsd:sequence>
                              <xsd:element maxOccurs="1" minOccurs="1" name="model" type="smtet:model"/>
                              <xsd:element maxOccurs="1" minOccurs="1" name="treatment" type="smtet:treatment"/>
                    </xsd:sequence>
          </xsd:complexType>
          <xsd:complexType name="treatment" mixed="true">
                    <xsd:sequence>
                        <xsd:element name="startTime" maxOccurs="1" minOccurs="0" type="xsd:dateTime"/>
                              <xsd:element name="timeUnit" default="UNIT" maxOccurs="1" minOccurs="1" type="smtet:timeUnit"/>
                              <xsd:element name="warmupPeriod" maxOccurs="1" minOccurs="1" type="smtet:period"/>
                              <xsd:element name="runLength" maxOccurs="1" minOccurs="1" type="smtet:period"/>
                              <xsd:element name="recordlog" maxOccurs="1" minOccurs="0" type="xsd:boolean"/>
                        <xsd:element name="numberofreplications" maxOccurs="1" minOccurs="1" type="xsd:positiveInteger"/>
                              <xsd:element name="properties" maxOccurs="1" minOccurs="0" type="smtet:properties"/>
                    </xsd:sequence>
          </xsd:complexType>
          <xsd:complexType name="properties">
                    <xsd:sequence>
                              <xsd:element name="property" maxOccurs="unbounded" minOccurs="1" type="smtet:property"/>
                    </xsd:sequence>
          </xsd:complexType>
          <xsd:complexType name="property">
                    <xsd:complexContent>
                              <xsd:restriction base="xsd:anyType">
                                        <xsd:attribute name="key" type="xsd:string"/>
                                        <xsd:attribute name="value" type="xsd:string"/>
                              </xsd:restriction>
                    </xsd:complexContent>
           </xsd:complexType>
          <xsd:complexType name="period">
                    <xsd:simpleContent>
                              <xsd:extension base="xsd:double">
                                        <xsd:attribute name="unit" type="smtet:timeUnit" use="required"/>
                              </xsd:extension>
                    </xsd:simpleContent>
          </xsd:complexType>
          <xsd:complexType name="model">
                    <xsd:sequence>
                              <xsd:element name="model-class" maxOccurs="1" minOccurs="1" type="xsd:string"/>
                              <xsd:element name="class-path" maxOccurs="1" minOccurs="0" type="smtet:classPath"/>
                    </xsd:sequence>
          </xsd:complexType>
          <xsd:complexType name="classPath">
                    <xsd:sequence>
                              <xsd:element name="jar-file" maxOccurs="unbounded" minOccurs="1" type="xsd:string"/>
                    </xsd:sequence>
          </xsd:complexType>
          <xsd:complexType name="replication">
                    <xsd:sequence>
                              <xsd:element name="stream" maxOccurs="unbounded" minOccurs="1" type="smtet:stream"/>
                    </xsd:sequence>
                    <xsd:attribute name="description" type="xsd:string" use="required"/>
          </xsd:complexType>
          <xsd:complexType name="stream">
                    <xsd:attribute name="name" type="xsd:string" use="required"/>
                    <xsd:attribute name="seed" type="xsd:positiveInteger" use="required"/>
          </xsd:complexType>
          <xsd:simpleType name="timeUnit">
                    <xsd:restriction base="xsd:string">
                    <xsd:enumeration value="MINUTE"/>
                    <xsd:enumeration value="HOUR"/>
                    <xsd:enumeration value="DAY"/>
                    <xsd:enumeration value="WEEK"/>
                    <xsd:enumeration value="YEAR"/>
                    <xsd:enumeration value="UNIT"/>
          </xsd:restriction>
          </xsd:simpleType>
</xsd:schema>
```